RecRoute
A Bus Route Recommendation System Based on Users' Contextual Information

Adriano de Oliveira Tito, Arley Ramalho R. Ristar, Luana M. dos Santos, Luiz Antonio V. Filho, Patrícia Restelli Tedesco and Ana Carolina Salgado
Center for Infomatics, Federal University of Pernambuco, Recife, Brazil

Keywords: Intelligent Transportation Systems, Recommendation Systems, Context-Aware Computing.

Abstract: Traffic has become an increasingly significant problem in the lives of citizens of major and medium sized cities. This has contributed to the inefficiency of public transportation, where one of the main issues to be tackled is the absence of relevant, timely information to users. In times where technology solutions for daily tasks are widely available, Public Transportation User Information Systems emerge as a possible solution to this issue, providing information to passengers and supporting their decision-making. This work aims to present a recommendation system for public transportation routes by bus, called RecRoute, that considers contextual information related to users, climate, time of day and traffic to recommend bus routes that are more adequate to the passengers' particular needs. The results of our experiment show that RecRoute was approved and its recommendations were well evaluated by the participants.

1 INTRODUCTION

Traffic in big cities is getting worse daily. With the increasing number of private cars on the roads, traffic jams have become more and more frequent. According to Zhang and colleagues (2011), widening streets, building overpasses or rotating cars are not viable alternatives for improving traffic. To this end, it is necessary that the government and the operators of public transportation provide better service, motivating users to use public transportation (Pilon, 2009). According to Cutolo (2003), the main barriers to the use of public transportation by bus lie in the lack of information about the services and low quality.

In this light, Intelligent Transportation Systems – ITS - aim to implement technologies to support the infrastructure and improve the quality of transport systems (Gómez et al., 2009). One of the subareas of ITS is called Advanced Public Transportation Systems - APTS, which consists of applications of ITS aimed at increasing the efficiency and safety of public transportation systems (Sussman, 2005). In the context of APTS, there are User Information Systems which constitute an important tool for communication between operators / managers and public transportation users. Such tools can provide passengers with information that meets their specific needs, expected arrival times of buses, the next vehicle to pass in the bus stop and route recommendations to users (Pilon, 2009).

Thus, in most cases, passengers of public transportation do not count with services that support them in choosing which bus and route take to reach their destinations. These services consider avoiding heavy traffic or accidents and taking into account passengers' current preferences. In our research project, called Ubibus (Vieira et al., 2011), we have proposed solutions to enable the use of mobile technologies and contextual information as support for public transportation passengers in large and medium-sized urban centers.

In this scenario, we present RecRoute: a context-sensitive route recommendation system for public transportation. RecRoute takes into account the context of the passengers as well as climate, time of day and current traffic situation. One major feature of our system is the acquisition and processing of contextual information from several sources.

The paper is organized as follows: Section 2 introduces concepts related to ITS, Context and Recommender Systems; Section 3 presents the Ubibus project; Section 4 presents RecRoute; Section 5 presents the experiments carried out and their results; Section 6 presents the related woks and Section 7 concludes the paper with some final

de Oliveira Tito A., Ramalho R. Ristar A., M. dos Santos L., Antonio V. Filho L., Restelli Tedesco P. and Carolina Salgado A..
RecRoute - A Bus Route Recommendation System Based on Users' Contextual Information.
DOI: 10.5220/0004866903570366
In *Proceedings of the 16th International Conference on Enterprise Information Systems* (ICEIS-2014), pages 357-366
ISBN: 978-989-758-027-7

remarks and future work.

2 BASIC CONCEPTS

This section presents some concepts of Intelligent Transportation Systems, Context and Recommender Systems.

2.1 Intelligent Transportation Systems

Intelligent Transportation Systems (ITS) investigate the application of technologies of communications, control, electronics and computer hardware and software to the surface transportation system in order to improve its quality (Sussman, 2005). Other environmental benefits can be obtained through the use of ITS, such as better rates of air pollution. With better traffic flow, greenhouse gas emissions will be reduced (Pilon, 2009).

There is also a sub-area of ITS called User Information Systems, that provides real time information regarding transportation availability in order to help users plan their trips.

Some solutions for User Information Systems already operate in developed countries, where the road system ensures efficient information about public transportation.

In towns with constant traffic congestion, and inefficient road systems, it is necessary to consider information beyond what is stored in databases (e.g. bus stop locations). The use of contextual information may help applications to become more adaptive for passengers, satisfying their needs and preferences.

2.2 Context

Computational Context can be seen as a set of conditions and influences relevant for an application and which make a situation unique and comprehensible (Brézillon, 1999). Context-Sensitive Systems are applications that adapt themselves without explicit user intervention, that is, they take into account information on the situation where the user is inserted to provide better services, thus increasing their usability and effectiveness (Baltrunas, 2011).

According to Vieira and colleagues (2009), the context of the interaction between an agent and an application to perform a task, is the set of instantiated contextual elements that are needed to support the current task. A contextual element is any data, information or knowledge that allows us to characterize an entity in a domain.

2.3 Recommender Systems

To Baltrunas (2008), Recommender Systems (RS) are powerful tools that can help the user face the problem of information overload by providing personalized recommendations. The main goal of recommender systems is to make suggestions for various services according to predetermined user profiles (Chorianopoulos, 2008).

RS can, for example, help the user to choose a travel plan, indicating places to visit options, hotels, and airlines, according to user preferences displayed on your profile.

There is still room for improvement. Traditionally, such systems are based only on user profiles or static variables, and do not take into account the changing circumstances that may influence the user's interests. Thus, the incorporation of contextual information in the recommendation process is highlighted in the literature as a possible extension to the traditional recommender systems (Adomavicius and Tuzhilin, 2008).

To Baltrunas (2011), a recommendation can often be more relevant if the context is known. For this reason, Context-Aware Recommender Systems - CARS, are gaining more prominence, and various approaches have been used incorporating the knowledge of the context. Thus, the new generation of recommender systems should explore context information to provide better recommendations.

3 THE UBIBUS PROJECT

The Ubibus project aims to facilitate the daily lives of public transportation users, providing intelligent access to information of this type of transport to the passengers in real time, based on dynamic contextual information related to their own means of transport (Vieira et al., 2011).

Figure 1 shows the Ubibus architecture. The *Data Level* is responsible for the management of information such as location, speed, bus route, locations of the bus stops, location of passenger, information of real-time traffic, maps and other. Information about jams is used to identify obstructions in the flow of traffic and their level, for example, slow traffic, moderate or congested.

The *Middle level* of the system consists of a middleware that facilitates communication and coordination among software components distributed transparently addressing the difficulties and complexities introduced by wireless communication and mobility, for example, access to applications for different types of devices.

The proposed middleware is multiparadigm and extensible, since it proposes to support a set of communication paradigms, and meet different types of applications, for example, mobile and web. To optimize the use of resources of the mobile devices integrated to the project, the middleware provides sharing and reuse of components. It is divided into three levels:

Communication Level: provides access to data, as well as real-time updates to managers, operators, users and drivers. The advances and standardization of wireless communication technologies, such as WiFi, Bluetooth, WiMAX, GPRS and 3G, allow communication of short and long reach, making it possible to develop applications for web, desktop, PDA, mobile phones, terminals and kiosks, for example, bus stops and bus stations;

Acquisition Level: is responsible for gathering contextual information from different sources, and for routing them to the *Data Level*. In Ubibus, contextual information may be obtained from sources such as social networks (e.g., Twitter and Facebook), GPS and monitoring cameras. Such information may be dynamic (e.g. the location of the bus) or inferred, for example, the discovery of a traffic jam and its intensity;

Processing Level: aims to carry out the treatment of the context information acquired from different sources in order to transform them into useful information for the applications to be developed. For example, regarding the source of GPS context, the *Processing Level* is responsible for receiving files with the locations and velocities of the bus at a given time frequency, and processing them so that they are

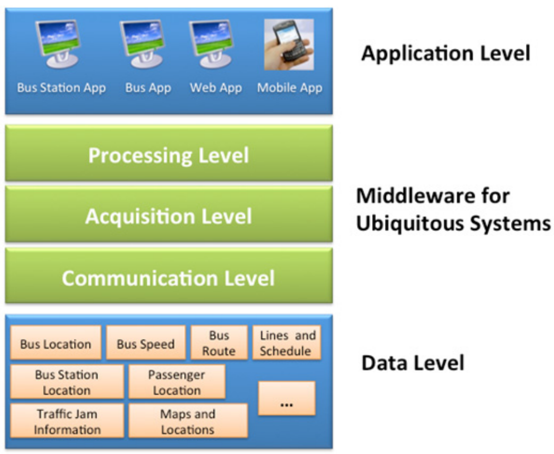

Figure 1: The Ubibus Architecture Overview (Vieira et al., 2011).

properly stored in the data layer.

The *Application Level* contains different types of applications developed. These applications fit the different platforms and devices such as web, desktop, PDA, mobile phones and displays.

4 THE RecRoute SYSTEM

The integration of various technologies has enabled ITS to evolve significantly. Consequently, there is an increasing demand for systems that are dynamic and context-sensitive. In particular, several route recommendation systems are being used as a way of providing users with timely, relevant information.

In RecRoute, contextual information allows the system to understand the user's preferences, weather conditions and traffic situation in order to adjust the results to the needs of passengers.

This section is divided into two parts. The first is the specification of RecRoute and the second part discusses how we have chosen the ranking algorithm used in the implementation of RecRoute.

4.1 The Specification of RecRoute

In this section, we present the modeling of contextual information, architecture and specification details of RecRoute, an application with mobile and web interfaces for recommending routes to public transportation users, based on contextual information of users, traffic and other urban factors that may affect it, such as the weather.

4.1.1 Modeling Contextual Information

To provide recommendations that are more appropriate to the needs of users, RecRoute uses static, dynamic and inferred contextual information. By considering the changes in contextual information, RecRoute can provide users with suggestions more targeted to their needs, considering that traffic has dynamic characteristics which incidents can, at any time, directly influence its status. Therefore, this section aims to show how the contextual requirements were elicited.

Initially we conducted a survey of contextual information about users' preferences that would possibly be useful for the passengers of public transportation during the recommendations. Such information are: shorter distance to be traveled, shorter travel time, lower ticket fares, bus exchange during displacement and distance to be traveled on foot.

To validate the preferences initially considered, we developed a questionnaire that was answered by 57 people, including students and users of public transportation by bus. Respondents were asked to rank the contextual information we provided and suggest any others they might have thought relevant.

By consolidating the results, we realized initially that preferences were considered “Very Important” and with “Average Importance” with special emphasis on the shorter travel time with almost 90% of relevance, the shorter distance to be traveled and bus exchange during displacement with 64.9% and 73.7% of relevance, respectively. It was also observed that none of the preferences initially suggested was classified as “Less Important” by most respondents, as shown in Figure 2.

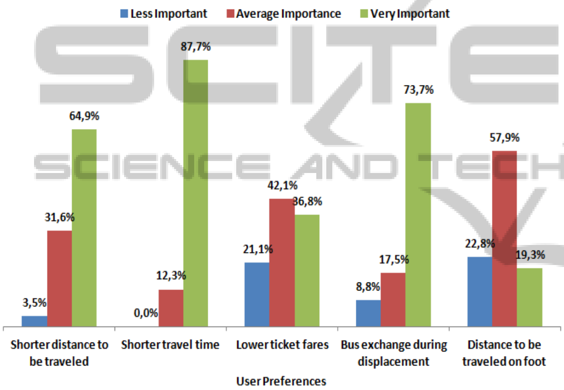

Figure 2: Relevance of User Preferences.

In addition to answers about the relevance of preferences, the waiting time at the bus stop was suggested by 60% of respondents as contextual information to be considered during the recommendations.

After analyzing the questionnaires, we were able to model more precisely the contextual information related to user preferences and other entities that are used by RecRoute. Figure 3 shows the model of contextual information used by RecRoute based on the UML metamodel proposed by Vieira (2009).

Among the contextual information modeled in Figure 3, RecRoute considers the following entities:

Passenger: relates to the requesting user and their preferences.

- **Location**: represents the geographic location of the user who is requesting the suggestion of routes.
- **Special Needs**: this information is important for the system because it informs if the user has some kind of disability such as mobility impairment (wheelchair), visual or hearing disability.
- **Distance Traveled on Foot**: This attribute tells what is the maximum distance in meters the user tolerates to walk.

Bus Exchange: it represents the user's preference for routes that have bus exchanges to be made along the way.

- **Type of Route Search**: it represents the user's preference for the type of search to be performed among the routes, for example, smaller distance, smaller fare or less time spent.

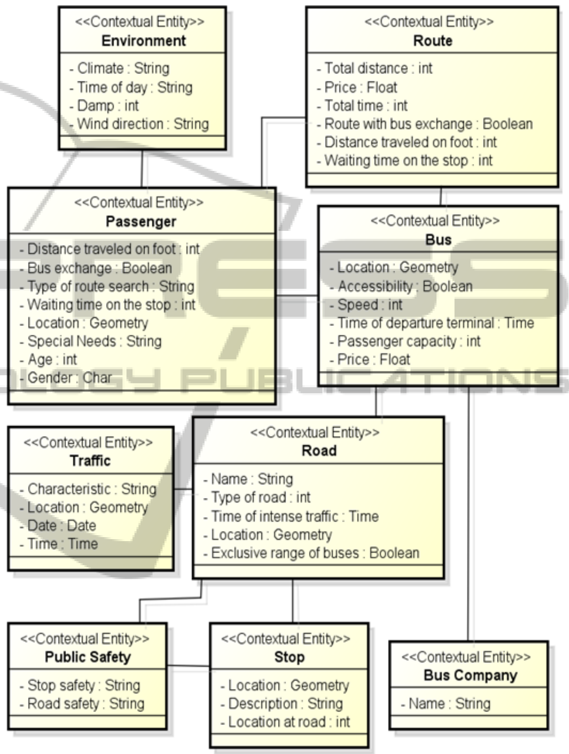

Figure 3: Modeling of Contextual Information.

- **Waiting Time on the Stop**: this information means the maximum time in minutes that the user will stand waiting at the bus stop.

Environment: this information concerns the environment in which the user is. For us the climate, time of day and other issues are important to the recommendations made by RecRoute because they can greatly influence the classification of routes and consequently the recommendations. We take into consideration**:**

- **Climate**: it represents the weather conditions at the time of the recommendation, such as rainy, sunny or cloudy.
- **Time of Day**: the information represents the time of day that the request is occurring as the morning (6:00 to 11:59 h), afternoon (12:00 to 17:59 h), night (18:00 to 05:59 h) or rush hour.

Route: routes may have many contextual information. The ones considered by RecRoute are:

- **Total Distance**: total distance to be traveled in meters.
- **Total Time**: total time spent to traverse the route.
- **Price:** total cost of the route.
- **Route with Bus Exchange:** it informs whether the user will need to exchange buses during their journey.
- **Distance Traveled on Foot:** indicates the distance to be covered on foot by the passenger.
- **Waiting Time on the stop:** this information relates to how long in minutes the user should remain waiting for the transport.

Bus: contextual information are as follows:

- **Location:** this information relates to the location of the bus at the time of the user request.
- **Accessibility:** it reveals whether the buses that travel the route have accessibility feature or not.
- **Price:** corresponds to the cost of the ticket.

4.1.2 RecRoute Architecture

RecRoute uses various features offered by the Ubibus middleware, such as the interfaces, the database, the service *Route Generation* that retrieves routes enriched for contextual elements of traffic according to the source and target passed by the user and the service *Climate and Time Information* which provides contextual information for climate and time used in the recommendations of routes. Figure 4 shows the RecRoute architecture.

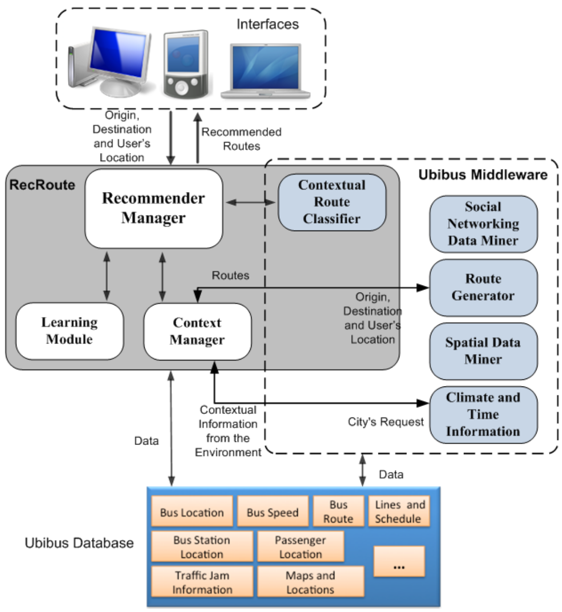

Figure 4: RecRoute Architecture Overview.

Four main components are present in the architecture of RecRoute: the *Recommender Manager* is responsible for the orchestration of the execution flow of the recommendation process; the *Context Manager* manages the acquisition of contextual information to be used by RecRoute, the *Learning Module* is responsible for the classification of routes and the *Contextual Route Classifier* responsible for ordering the routes. This last component is part of the Ubibus Middleware and also offers its services to other applications. Routes are classified by their characteristics considering the point of origin and destination chosen by the user.

- ***Interfaces:*** The users communicate with RecRoute through its Web and Mobile interfaces. The mobile interface has the benefit of being portable, allowing the user to make a decision en route, even if not at the bus stop or at home. This is similar to the web Interface adapted for mobile devices and also has the possibility of using the georreferenced position of the user device more precisely. Figure 5 shows screenshots of the mobile interface.

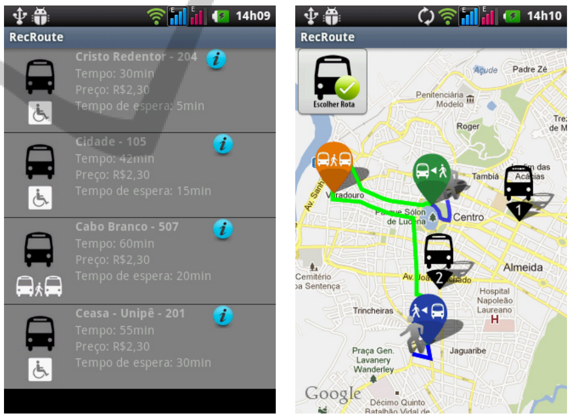

Figure 5: Example of the Mobile Interface.

- ***Recommender Manager:*** The Recommender Manager is the core component of RecRoute and manages all application actions, for example, the calling of other components of the architecture. It is through this element that all actions performed by the users through the interfaces access to system functionality.
- ***Context Manager:*** The *Context Manager* is responsible for managing all contextual information that will be used for the recommendation. This component communicates with Ubibus middleware services to obtain the necessary data such as the initial set of routes provided by the *Route Generator* for the origin and destination as well as climate and temporal information.

- ***Learning Module:*** This module manages the function used for the classification of routes through supervised learning (training) incrementally. In this process, we have formed a set of training records, called the training set, containing information about the user's preferences, climatic and temporal information, two options of the routes, and their associated class labels, in this case the best route between options presented (described in section 4.2), and submitted them to a *Naive Bayes* classifier (Friedman, 1997), generating a function.

The function, which results from the learning of training records for the *Naive Bayes* classifier, is stored in a structure in the database. Each function is unique to the points of origin and destination provided by the user upon request. Thus each request that has the same origin and destination will use the same function for classification.

- ***Contextual Route Classifier:*** The *Contextual Route Classifier* is the component responsible for ordering routes using the function produced by the Learning Module. Routes are classified by their features, user preferences and chosen points of origin and destination.

The classification of routes is carried out by grouping them in pairs and using the expression N*(N-1), where N is the initial number of routes, resulting number of pairs.

For example, Figure 6 shows the formation of pairs of routes for classification with a group of 4 routes. For this case 12 pairs of routes are formed, these pairs are repeated alternating positions between routes (first and second route).

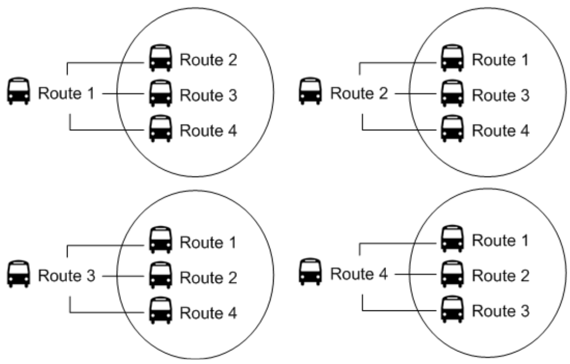

Figure 6: Illustration of Grouping Routes to Classification.

This way of grouping in pairs to pass through Naive Bayes was chosen because we believe that the repetition of pairs alternating positions minimizes the margin of error of classification. After the formation of these pairs the classification starts.

After evaluating all pairs by sorting function and the consolidation of scores, all routes are ordered according to the score obtained by each route during evaluations.

- ***Middleware Services Used by RecRoute:***
  - *Route Generator*: This component generates a set of routes according to points of origin and destination indicated by the user. Contextual traffic elements such as traffic jams, hours of heavy traffic, accidents, floods, social networking information and other information are used to generate the routes.
  - *Climate and Time Information*: This component is responsible for providing information about the climate and weather that are used by RecRoute. To obtain this information the *Context Manager* informs the city from where comes the recommendation and receives the climate situation and time information in the locality.
- ***Database:*** The database used by RecRoute is shared with other applications and services of the Ubibus Project. This database stores all obtained contextual data about traffic, buses, routes, bus stops, and passengers, among others.

4.2 Choice of the Classification Algorithm

As described in Section 4.1.2, learning functions were created through *Naive Bayes* algorithm to assist in the route recommendation. These functions are generated by processing a set of registers called the training set.

The *Naive Bayes* algorithm is also used to update functions, according to the user's choices among the recommendations made by the system. Thus, we can see that the choice of the classification algorithm is very important in the scope of this project.

The experimental setting adopted to compare the performance of algorithms aimed to verify which of the algorithms - *C4.5* (Quinlan, 1993), *Naive Bayes*, *Multilayer Perceptron* (Mcculloch and Pitts, 1943) – would perform better in the scenario of our application.

In this experiment, the implementation of machine learning algorithms was done with API Weka (Waikato, 2010). Weka uses a file called ARFF (Attribute-Relation File Format) file that contains a set of records used in the learning task. In the case of this study the ARFF consists of contextual information of passengers, the environment and bus routes.

The ARFF file consisted of records where each row represents a comparison between two routes, according to user preferences and climate and temporal data. Figure 7 illustrates the format of records contained in the ARFF file, comprising five-parts:

The first represents the contextual information of users, including special needs (if any), distance to be traveled on foot in meters, preference by bus exchange, search type of route (smaller distance, less time or smaller price) and waiting time at the stop in minutes;

The second represents the contextual information of climate and time when the route was requested;

The third and fourth parts comprise the information of the two routes being compared: total distance to be traveled by the route in meters, total price of route, total time in minutes, route with bus exchange, accessibility on buses, distance to be traveled on foot and waiting time at the stop in minutes;

The last part of the records corresponds to the choice made by the user, considering their preferences, the setting and the characteristics of the two routes displayed. In Figure 7 we see that the route chosen was route 2.

To obtain the data to train the algorithms, we developed a Web page, where users visualized an environment (weather conditions and climate) and two routes randomly generated. Then they informed their preferences and chose one of the two routes previously displayed, thus creating a record for the training set. The comparison was always made between two routes

Access to the page was available during two weeks. In this period 742 records were collected. After the acquisition of the records with the contextual information of users, we defined some evaluation metrics for the experiment, taking as parameters the scene usage by RecRoute:

- **Percentage of Correct Classifications:** The percentage of success is provided by WEKA after performing the training of the records contained in the ARFF file and corresponds to the degree of efficiency of the algorithm tested in predicting correctly the route preferred by the user;
- **Total Time to Construct the Function:** This measure relates to the total time taken to construct the function that is used to rank the routes;
- **Time Required for the Classification of 1 record:** To recommend routes RecRoute assigns scores from the rank held by the function of records with structure similar to ARFF file. So we measured the time required for classifying each record.

To carry out this experiment, we have used a computer with Intel® Core™ 2 Duo Processor P7550 (3M Cache, 2.26GHz, 1066MHz FSB), 4GB of RAM, x64 Windows 8 operating system and Java SE version 1.7.0_09 x64. The results obtained during the experiments are shown in Table 1.

Table 1: Results Comparison of Algorithms.

Metric / Algorithm	*C4.5*	*Multilayer Perceptron*	*Naive Bayes*
Percentage of correct classifications	82.34	84.77	85.25
Total time to construct the function (in seconds)	0.05	9	0.01
Time required for classification of 1 record (in seconds)	0.0015	0.002	0.006

With respect to the time required for the construction of the function, we note that the worst performance was obtained by *Multilayer Perceptron*, with results well above the rest. With respect to the time required to classify one record the algorithms have obtained a similar result with the *C4.5* performing better.

According to the data presented and considering the unique case of the proposed recommendation system, we conclude that the *Naive Bayes* algorithm had the best performance, it had the best success rate and less time to generate the function that classifies routes. Despite presenting the worst time for the classification of individual records, the difference for the other algorithms (in seconds) is considered small. So, we decided to use the *Naive Bayes* for implementing RecRoute.

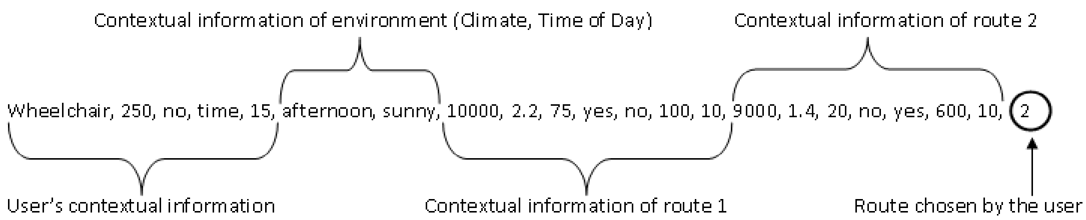

Figure 7: Illustration of a Record of the ARFF File.

5 EXPERIMENTS

The experiments performed had the following objectives:

Assess whether the recommendation process is performed as expected;

Assess whether the routes recommended, actually comply with the needs of users and check the position of the route chosen by the user among the list of routes displayed;

Gather information for possible future improvements.

The preparation of the experiment consisted of two steps. The first step was the implementation of two prototype interfaces (web and mobile), through which the experiment was conducted. So, the user could access the system, informing their preferences and make requests for routes to the points of origin and destination already pre-set.

The second step was the construction of two experimental scenarios, considering the points of origin and destination pre-established. Considering that the service *Route Generator* was still under development, routes to the experimental scenarios were generated with actual data stored in the database (e.g. points of stops and stretches to be traveled) and other simulated information (e.g. waiting time on the stop). For the configuration of experimental scenarios, real points of origin, destination and bus lines stored in the database were used.

The experiment was performed by a group of 20 participants, residents of the city of João Pessoa that use public transportation by bus. The experiments were performed individually with each volunteer having agreed with the experimental scenarios proposed. At the beginning of the experiment the objectives, RecRoute characteristics, how to use it and the experimental scenarios proposed were explained to each participant.

During the execution of the experiments, participants used the experimental scenarios and, after viewing the suggested routes, they analyzed and elected the best route for them according to the proposed scenarios.

Participants were asked to answer questions related to the objectives of the experiments. The responses were collected and analyzed for the two experimental scenarios and will be shown below.

About the quality and correctness of the recommendations made by RecRoute, participants were asked whether the route indicated in the first position of the list suited their needs and if the most frequent of this list would be chosen. The percentage of correct answers for the routes that appear in the first position of the ranking was 75% for the experimental scenario 1 and 90% for the second scenario.

This difference might be related to the lower amount of routes for scenario 2. In other cases the route chosen by the participants was listed as the second option and was chosen by the participants because of the shorter distance to be traveled on foot in relation to the first.

The routes suggested by RecRoute after the 2nd option were not chosen by the participants at any time. Thus, the routes chosen were always in the 1st or 2nd positions of the list. Figure 8 illustrates the percentage of correct responses in each experimental scenario.

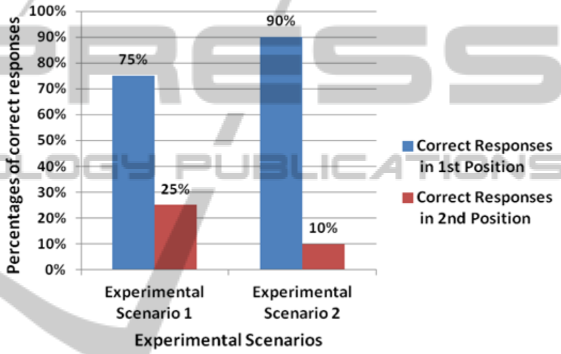

Figure 8: Correct Responses X Experimental Scenarios.

Some improvements were proposed by the participants, as follows:

1. Adapt the interface for the visually impaired. The inclusion of audible and vibrating alerts could be a good alternative, as well as voice recognition;

2. Include other information that could influence the ordering of routes and preferences of users. Some information suggested were how full the bus is, hazard risk along routes, bus stops and also where to get in and to get out the bus and bus exchange;

3. Consider the context of the user to display the path to be traveled on foot, showing paths with greater accessibility in case of wheelchair and visually impaired users;

4. Send clue when the bus is approaching places that might require user action, e.g., boarding.

6 RELATED WORK

In the literature one can find many studies in the field of ITS and UIS, but only some of them make use of recommendation and/or display of public transportation routes by bus.

In this section, we analyzed some studies found that exploit recommender systems and/or display routes to users, and that are thus related to the RecRoute

In order to facilitate the analysis of the related work we adopt some criteria for comparison, described below:

- **Use of Contextual Information**: The use of dynamic, static and inferred contextual data and their interactions enable the application to adapt to a given situation or provide more relevant services;
- **Communication Interface with Users**: The communication interface between recommender systems and their users can be a very important factor to the effectiveness of the system. Some of these systems allow access from anywhere and in many ways, allowing users to have real-time information, adapted to the dynamics of transit;
- **User Preferences**: For recommender systems, user preferences are key to providing more personalized and adapted to the same information;
- **Users' History of Usage**: The historical usage data of users can generate useful information for recommendation system, because through them we can infer important contextual data, such as preferences for using the system and thus provide more relevant recommendations;
- **Field of Use Only for Public Transportation by Bus**: this criterion evaluated whether the application is directed only to public transport by bus.

Table 2 shows some studies, analyzing them according to the criteria above.

We observed that, some studies shown in Table 2 are not totally directed to public transportation by bus and few consider all types of contextual information (Static, Dynamic and Inferred) when making suggestions.

It was also possible to note that the vast majority of them do not incorporate contextual information about the climate, time and user context, eg, preferences and historical usage. Thus, we conclude that these applications provide features not suited to passengers.

The RecRoute has many interfaces, is totally directed to public transportation, includes the capture and processing of dynamic, static and inferred data of users, traffic and environment, in order to provide information and bus routes more realistic and adapted to the real needs of the users of urban public transport by bus.

7 CONCLUSIONS AND FURTHER WORK

Currently, Intelligent Transportation Systems turned into a very viable and attractive alternative to solve overcome challenges in the transportation of the large cities. This evolution is partly due to the accelerated growth of Information and Communication Technologies.

Table 2: Comparing the Related Work.

Studies / Criteria	Use of contextual information	Communication interface with users	User preferences	History of use of the user	Field of use only for public transportation by bus
OneBusAway (Ferris et. al., 2009)	Static, Dynamic and Inferred	Mobile, Web e SMS	No	No	Yes
Bus Catcher (Bertolotto et. al., 2002)	Static and Dynamic	Mobile	Partly	No	Yes
Traffic Information System (Hoar, 2010)	Static and Dynamic	Mobile e Web	No	No	Yes
PECITAS (Tumas and Ricci, 2009)	Static	Mobile	Partly	No	No
ANTARES (Bastos and Jaques, 2010)	Static	Web	No	No	Yes
UbibusRoute (Lima et al., 2012)	Static and Dynamic	Mobile	Partly	No	Yes
RecRoute	**Static, Dynamic and Inferred**	**Mobile e Web**	**Yes**	**Yes**	**Yes**

This development contributes to the increasing use of computer systems in almost all areas of human activity. Thus, there is a rising demand for dynamic, context-sensitive systems. The use of this type of application in providing information to users of urban public transport can provide greater attraction and loyalty to the service.

This work has presented the RecRoute route recommendation system for users of public transportation by bus, able to process static and dynamic contextual information, of the users, bus lines, climate, time and traffic, providing more fitting recommendation for the passengers.

This solution differs from other related work by the use of dynamic contextual information from various sources by using different devices to enable ubiquitous and context sensitive use being directed to public transportation passengers. RecRoute is integrated to the Ubibus project and is one of its applications.

Future works are related to the suggestions made by participants of the experiment and more experimentations, aiming the improvement of RecRoute as follows: develop versions for other operating systems of mobile devices, in addition to Android, such as IOS and Windows; calibrate the importance of contextual information used by the application; and evaluate other algorithms that can improve the recommendations provided.

REFERENCES

Adomavicius, G., Tuzhilin, A., 2008. Context-Aware Recommender Systems. In *Conference on Recommender Systems*, pp. 335-336.

Baltrunas, L., 2008. Exploiting Contextual Information in Recommender Systems. In *ACM Conference on Recommender Systems*, pp. 295-298.

Baltrunas, L., 2011. *Context-Aware Collaborative Filtering Recommender Systems*. 172 f. Tese (Phd. Thesis in Computer Science). University of Bolzano. Bolzano, Italy.

Bastos, R., Jaques, P., 2010. Antares: Um sistema web de consulta de rotas de ônibus como serviço público. *Revista Brasileira de Computação Aplicada*, vol. 2, pp. 41-56.

Bertolotto, M., O'Hare, G., Strahan, R., Brophy, A., Martin, A., McLoughlin, E., 2002. Bus catcher: a context sensitive prototype system for public transportation users. In *2th International Workshop on Web and Wireless Geographical Information Systems (W2GIS)*, Singapore, pp. 64-72.

Brézillon, P., 1999. Context in Artificial Intelligence: IA Survey of the Literature. *Computer & Artificial Intelligence 18*, pp. 321-340.

Chorianopoulos, K., 2008. Personalized and mobile digital TV applications. Multimedia Tools and Applications. In *Kluwer Academic Publishers*, v.35, n.2, pp. 1-10.

Cutolo, F., 2003. Diretrizes para sistema de informação ao usuário. In *3th Seminário Internacional PROMOTEO*, Porto Alegre, RS, Brazil.

Ferris, B., Watkins, K., Borning, A., 2009. Onebusaway: A Transit Traveller Information System. In *Mobicase*. San Diego, USA, pp. 92-106.

Friedman, D., 1997. *Machine Learning 29*, pp. 131-163.

Gómez, A., Diaz, G., Bousetta, K., 2009. ITS Forecast: GIS Integration with Active Sensory System. In *Information Infrastructure Symposium*, pp. 1-6.

Hoar, R., 2010. A personalized web based public transit information system with user feedback. In *13th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, Ilha da Madeira, Portugal, pp. 1807-1812.

Lima, V., Magalhães, F., Tito, A. O., Santos, R., Ristar, A., Santos, L., Vieira, V., Salgado, A. C, 2012. UbibusRoute : Um Sistema de Identificação e Sugestão de Rotas de Ônibus Baseado em Informações de Redes Sociais. In *8th Simpósio Brasileiro de Sistemas de Informação*, São Paulo, Brazil. pp. 516-527.

Mcculloch, W. S., Pitts, W., 1943. *A Logical Calculus of the Ideas Immanent in Nervous Activity. Bulletin of Mathematical Biophysics*, n.5, p.115-133.

Pilon, J. A., 2009. *Sistema de Informação ao Usuário do Transporte Coletivo por Ônibus na Cidade de Vitória-ES*, Universidade Tecnológica Federal do Paraná – Ponta Grossa-PR, Brazil.

Quinlan, R., 1993. *C4.5: Programs for Machine Learning*, Morgan Kaufmann Publishers, San Mateo, USA.

Sussman, J., 2005. *Perspectives on Intelligent Transportation Systems*. New York, USA: Springer.

Tumas, G., Ricci, F., 2009. Personalized mobile city transport advisory system. In *Information and Communication Technologies in Tourism*, Amsterdam, Nestherlands, pp. 173-183.

Vieira, V., Tedesco, P., Salgado, A. C., 2009. *Modelos e Processos para o Desenvolvimento de Sistemas Sensíveis ao Contexto*. Jornadas de Atualização em Informática. Porto Alegre, RS, Brazil, pp. 381-431.

Vieira, V., Caldas, L., Salgado, A. C., 2011. Towards an Ubiquitous and Context Sensitive Public Transportation System. In *4th International Conference on Ubi-media Computing*, São Paulo, SP, Brazil, pp 174-179.

Waikato, 2010. *Weka 3 – Machine Learning Software in Java*. URL: http://www.cs.waikato.ac.nz/ml/weka.

Zhang, J., Wang, F., Wang, K., Lin, W., Xu, X., Chen, C., 2011. Data-Driven Intelligent Transportation Systems: A Survey. *IEEE Transactions on Intelligent Transportation Systems*, pp. 1624-1639.